Système d'exploitation II

TP 4 : Processus zombies

Un processus Zombie est l'état d'un processus dont le père n'a pas encore pris connaissance de sa terminaison. Il se caractérise dans le résultat affiché par la commande ps par la valeur Z de l'état, 0 pour l'espace mémoire et sur de nombreuses versions 'defunct' dans le champ commande. Pour éviter l'état zombies on utilise les primitives wait et waitpid

Ce sont ces primitives et elles seules qui permettent l'élimination des processus Zombies et permettent la synchronisation d'un processus sur la terminaison de ses descendants avec récupération des informations relatives à cette terminaison.

L'appel système wait() permet à un processus appelant de suspendre son exécution en attente de recevoir un signal de fin de l'un de ses fils.

Syntaxe : int wait(status);
int *status;

Les principales actions de wait sont:

- Recherche de processus fils dans la table des processus : si le processus n'a pas de fils, le wait renvoie une erreur, sinon elle incrémente un compteur.
- S'il y a un zombie, il récupère les paramètres nécessaires et libère l'entrée correspondante de la table des processus.
- S'il y a un fils mais pas de zombie, alors le processus se suspend (état endormi) en attente d'un signal.
- Lorsque le signal est reçu, la cause du décès est stockée dans la variable "status". 3 cas à distingués : processus stoppé, processus terminé volontairement par exit, processus terminé à la suite d'un signal.
- Enfin le wait permet de récupéré le PID du processus fils : PID=wait(status).

La primitive **waitpid** permet de sélectionner parmi les fils du processus appelant un processus particulier.

- Exécuter le programme suivant :

```
#include <sys/times.h>
        #include <stdio.h>
        main(){int PID, status;
                if (fork()==0) {printf("processus fils %d\n", getpid());
                        exit(10);}
                printf("processus père %d\n", getpid());
                for (;;);                        /* le processus père boucle */
        }
```

- Taper la commande ps –a, qu'est ce que vous remarquez ?
- Dans ce cas c'est qui le processus zombies
- Généralement qu'elles sont les causes de la création des processus zombies.

- Exécuter le programme suivant :

```
#include <sys/times.h>
        #include <stdio.h>
        main(){
                int PID, status;
                if (fork()==0)
                        {printf("processus fils %d\n", getpid());
                        exit(10);}
                PID=wait(status);
                printf("processus père %d\n", getpid());
                printf("sortie du wait \n ");
                sleep(15);
                        /* fils est bien terminé père toujours en place signal et infos
              reçus */
                printf("PID = %d status = %d\n", PID, status);
                exit(0);
        }
```

- Expliquer la valeur de la variable status.
- C’est quoi le rôle de la fonction wait dans ce programme.
- Taper la commande ps –a, qu’est ce que vous remarquez ?